Abstract

Two methods generally use for verification of module at System on Chip (SOC) level are Formal Verification and Functional Verification. In the first phase of report evolution of the verification is explained which start from HDL task based strategy which is replaced by the object-oriented verification to the Random generation to the test bench tool and finally the complete test bench automation system. The next part gives brief introduction about verification categories at the Free scale which are Functional block, integration and processor design. The next phase gives the information about the Formal and Functional verification at SOC level. After that the RAM module is discussed which includes the features and the configuration of the RAM module at SOC level. The next part is SOC verification which includes the common use scenario in SOC verification and when we say that SOC verification is complete. The ending part give the knowledge about how to standardizing C programming conventions which includes common type definition, register/memory access macros and interrupt service routine. Second part of the report discuss about register automation of SOC. Basically it's a Perl script which takes input from XL sheet and generate verilog and c test cases. This test cases are use to verify memory map, reset value and read-write accessibility of configuration registers available in SOC. It supports 8, 16, 32 bit architecture SOC.